E-lection

Online election manager/publisher/analyzer

Requirements Specification and Analysis

Version 0.1

17.11.2017

Abdulwahab A. Sani

Berkay Yılmaz

Can Utku

Recep İlkay Depe

Prepared for

SE301 Software Engineering



< E-lection >

Table of Contents

**REQUIREMENTS ANALYSIS DOCUMENT[1]**

**1. Introduction**

The purpose of this system is to provide an efficient voting platform for various institutions ranging from schools to offices. The proposed system was designed so as to enable users to hold an election with ease. The driving motive behind the development of the system was a growing demand for a platform that will help with the choice of a person with majority votes fairly while at the same time the system should be user-friendly and need minimal training/knowledge to be able to use.

**1.1.Purpose of the System**

This system was designed with the aim of providing easy, credible and fair elections for users who want to use a vote to choose a person with the highest number of voters thereby declaring the person the winner. It makes the whole process an easy affair without too much hassle as all the users have to do is sign up and follow a few easy steps.

**1.2.Scope of the System**

The system provides service to mostly the person who in charge and also universities, government departments and so. Thus, there are some common functionalities for both admins who pay for the service and the users, also some different functionalities for the both too.

The system provides many services to the admins of the system like; Admins can create new election to select a person to a specific position by the votes of people that affected by the results or if admin eliminated a lot of candidates and can’t decide between several of them, then they can create an election to choose it. They can see the analysis of the elections, they can see the changes between votes between 2 elections and they can manage the system according to themselves.

The system provides many services to the Candidates like; they can vote for themselves or somebody else in the same election. They can see who is winning live before the election ends and they can prepare for a demonstration about what they want to achieve if they win the election to voter’s to see while they are voting.

The system provides some services to regular Users; if they are allowed, they can vote in private elections and they can vote in public elections(regional, state etc…). They can follow the election by time going and after the election ends, they can see full analyzed results of the election they participate.

**1.3.Objectives and Success Criteria of the Project**

Our objective is to grant users the ability to conduct elections easily without having to go through unnecessary hardship. This will be a base system for future online elections and a good choice for today's business elections, to get rid of a lot of paperwork and man-hours spent.The success criteria for the project would be:

* users’ satisfaction
* the rate of usage
* reported successful elections
* the popularity of the system
* usefulness

**1.4.Definitions, Acronyms, and Abbreviations**

RAD: Requirements Analysis Document

**1.5.Overview**

This document contains descriptions of the functionalities of the proposed system. Functional and non-functional requirements of the system are explained in detail. There is also a description of the use cases and scenarios used. Also included is the schedule for when the project is estimated to be completed also known as Gantt Chart

In Current System section of our RAD documentation, we talked about functions and features of current Election system which is not a Computer based system. We mentioned disadvantages of current system such as too long time and people need to work for and some times it can be very frustiating enough to people not to vote. and also we explained that the current system does not work easy and properly. In addition, we represented that current Election system is not futuristic , not stable ,not fast and requires a lot of man-power.

In Proposed System section, we talked about our new Election system’s advantages and modifications. For example, we defined that needed man-power, time consuption and the work to do by voters less than previous system or users can access whatever they want with less click. In addition, we explained that our new Election system will be online and more efficient by comparison current paper based election system.

In Overview section, we described overview of features, functions and details of our online election System. In addition, we explained usage and benefits from users’ perspective.

In Functional Requirements section, we talked about functions and features of Our online election system. And also, we described functions of our project from voters’, cancidates’ and System admins perspective. Moreover, we mentioned interaction between our project and user and their environments.

In Nonfunctional Requirements section, we mentioned nonfunctional parts of our new system such as usability, performance, reliability or availability and so on. We described all of nonfunctional requirements of our project.

In System Models section, we described scenarios and use cases of our project. We defined scenarios, actors, and use cases’ flow events and so on in this part.

In Object Model section, we explained class diagrams of our system and we defined relationships between classes of our project.

In Dynamic Model section, we mentioned sequence diagram that is to say we explained methods and functions and their operations with actors of our project.

In Glossary section, we described all things of our project system, it can be called as project dictionary. We explained all words in RAD documentation.

**2. Current System**

The current election system is used generally around the world is on paper election method that requires voters to come to a specific place to vote and besides regional elections for examples in schools, this can be a really big problem.For example last year there was an election in our schools which get %40-50 participation percentage which is very very low. With that system. If we want to continue with school example the paperwork needs to be done by the school and candidates and when the election is going through there are a lot of people working in the process to keep the election safe and fast as possible it. After the election is finished there is a calculating votes process and you can’t be sure if it's accurate or not fully. When the election is happening voters need to go to a specific place that is mostly not close to their house or they don’t want to enter the queue for voting. Moreover, current election system is functional but not optimal.

**3. Proposed System**

The new system to be designed will enable voting by allowing candidates to sign up to an online system so as to be able to be elected and voters will also sign up to vote for the

candidate of their choice. This eliminates the inconvenience of having to be physically present before a voter can vote. It will also increase participation in the elections and we can be looking at an improvement of up to 80% or more (from a poor 40-50%). Our proposed system will make elections tidier because there is minimal or no paper work thus, our system

improves accuracy and reliability of elections. Candidates have a unique ID which will be

important when the voter is choosing for his choice of candidate to vote for. Election rigging is an unfortunate but common practice in many elections in the current system however, our system will prevent and that is a significant and welcome upgrade. The system ensures that the elections aren’t rigged (that is no voter is allowed to vote for the same candidate more than once) by asking for the voter’s user ID before they make their vote. So, when a voter with the same user ID tries to vote again for a candidate with the same candidate ID, the system rejects the vote as invalid. Counting the ballots after the elections can be stressful work for the person tasked with the job. As mentioned our system has minimal to no paper work. Votes are automatically tallied and viewed at the end of the elections. This not only improve accuracy significantly but also saves a lot of time which is also very important.

**3.1.Overview**

E-lection is an online election system that can be used by everyone from local school elections to regional elections by in three status which is ; voter, candidate and admin. Voters use E-lection to select their favorite candidate that will eventually affect them in some way in the future.They can see the candidates propogandas and they can see the result of an election detailed. In addition voters vote by using E-lection. Candidates use the system to reach their goals and become selected for spesific position , that could be a school presidency or that could be an upgrade in their current office. They can vote for themselves or others in the elections they are allowed to vote. They can use the system for publish their propoganda and so on. The E-lection can be used anywhere via internet browser. E-lection has common and different functionalities for all of the users. The users of the system are Voter , Candidate and admin. They have a lot of functions in common but several differances such as , voter and candidate can vote but candidate has approvel to see the election analytics before election ends. On the other hand only admin can create an election, and the candidates are choosed by previous conclusions or by just saying that you want to enter as a candidate , if admin approves the request than you are a candidate for that spesific election or you need to accept the invitation for candidateship. In E-lection voters can vote , change their profile inforrmation , look into the finished election analysis or get to know better the candidates. Candidtes can do the same actions as voter and also they can follow the election alaysis as election goes , they can partitcipate an election and create their own page to make propoganda. In addition this election system controlled by admin and users are free to explore. This will be a system which removes the all manwork, hours of procces and safety problems.

**3.2.Functional Requirements**

E-lection is an online election manager/publisher/analyzer system platform for anyone who wants to make an election safe , stabilized and fast.On voter’s side E-lection has several functions such as , they can participate public elections or if they are allowed they can vote in private elections to support their candidate , they can see the analyzed report of an election that they are participated which ended. They can see the candidates profile, their profile or reports about the election. In addition they can cast vote in an election , they can get information about the candidates and update their profiles.

On candidates side, E-lection has several different functions besides the voters such as; They can follow the election more detailed , they can run their campaign on the website as well and they can vote for themselves. Moreover they have more controll over the system than voters.

Admin is the one who can create an election and implement the candidates to the system to vote. Admin can publish election, admin can update,delete or promote candidates accounts through the system.

The main and important function of E-lection is candidateship procces. Admins determine the candidates and there is a follow up procedure need to be complete to be an candidate.

**3.3.Nonfunctional Requirements**

*Usability*: For a person who needs to vote should make fourteen clicks to register (including clicks made for login, approve and start registration).

*Reliability*: The system must be running 100% of the time when there is an election.

*Performance*: The system must allow least 500 parallel users.

*Supportability*: The system must be able to be maintained and changable easily.

*Implementation*: The system will be implemented on Visual Studio platform. In addition, C# will be used as the programming language and ASP.NET,PHP will be used in the process. User Interface should be web-based (accessible via WWW Browser). Can be changed in the future.

*Interface*: There is no external systems or legacy systems to work with. Thus, there is no interface requirements.

*Packaging*: No constraints on the actual delivery of the system are determined. Thus, packaging requirements will be decided in the future.

*Legal*: The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and no infringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with the software or the use or other dealings in the software.

**3.4.System Models**

**Scenario 1:**

**Scenario name: Sign-Up as Voter**

**Participant actor instances: İlkay: Voter**

The flow of events:

1. İlkay wants to vote in an election.In order to do that he has to sign-up to the system. When he

entered the site he clicked green sign-up as voter button which is located to the right top of the

screen.

2. The upcoming page consists of required information. So, Kristy fills the information which are:

name, surname, e-mail address, organization name(not necessary), password and password

confirmation. Ilkay filled blanks and agrees on terms & conditions then press the sign-up button.

3. After clicking the sign-up button if any error occurs user is informed and wrong information are

shown with red color. After re-filling, Elkay clicks the sign-up button again.

4.If any error occurs event #3 is followed.If done, the new page informs the user to activate his

membership via. e-mail address. Activation link is sent to the user's given mail address.

5. Ilkay opens corresponding e-mail and opens the activation link given.The new page informs

him that his membership is activated and gives him a user ID.

**Scenario 2:**

**Scenario name: Sign-Up as Candidate**

**Participant actor instances: Kristy: Candidate**

The flow of events:

1. Kristy wants to participate/vote in an election.In order to do that she has to sign-up to the

system. When she entered the site she clicked green sign-up as voter button which is located

right next to the sign-up as a voter button.

2. The upcoming page consists of required information. So, Kristy fills the information which are:

name, surname, e-mail address, organization name(not necessary), password and password

confirmation. Kristy filled blanks and agrees on terms & conditions then press the sign-up button.

3. After clicking the sign-up button if any error occurs Kristy is informed and wrong information are

shown with red color. After re-filling, Kristy clicks the sign-up button again.

4.If any error occurs event #3 is followed.If done, the new page informs Kristy to activate her

membership via e-mail address. Activation link is sent to Kristy’s given mail address.

5. Kristy opens corresponding e-mail and opens the activation link given.The new page informs

her that her membership is activated and gives her a candidate ID.

**Scenario 3:**

**Scenario name: Log-in**

**Participant actor instances: ilkay: Member of the site**

The flow of events:

1. ilkay opens the election site in order to log in.

2. The e-mail address and password fields are filled. He presses the sign-in button.

3. Both password and email match, there is no error so logging in process is successfully

completed.

4. System briefly asks ilkay if he wants to be remembered next time he attempts logging in.

5. He chooses yes.

**Scenario 3 Extensions:**

2a. ilkay inputs wrong email or password so logging in the process fails. He is asked to recheck the

information he provided.

2b. ilkay inputs wrong email or password so logging in the process fails. He is asked to recheck the

information he provided. He realized he has forgotten his password. The system guides him to reset the password by sending a reset link to his attached email address.

5. He chooses no so next time he tries logging he has to manually input all his information.

**Scenario 4:**

**Scenario name: Create an Election**

**Participant actor instances: Max: Admin of the site**

Flow of events:

1. Max is a businessman who wants to hire a chief officer for his company. He had conversation

with 5 possible people and he is not sure which of them is the best option.In order to select best

chief officer for his company, he wants to do an election with public relations department of his company.

2. Since Max is already member of the site he log on to the system with his email and password.

3. He opens the menu which located left side of the screen, and clicks create an election bar.

Next page asks him to fill blanks which are Election Name,Time period of the election,

organisation name(not necessary) then clicks the next button.

4. If organisation name are filled the system automatically adds the voters committee which is

(in this case) workers of public relation department of company.

5. On the next page, Max has to add candidates. Max can add up to 10 candidate by clicking plus

icon. When icon clicked pop-op screen will appears and following informations needed:

candidate name, surname, candidate ID, photo of the candidate, candidate CV's(In this

case).When steps are done Max clicks done button and system adds the candidate then pop up

screen gone. Max do these process 10 times to add 10 candidates.Then clicks start election

button.

6. When election started, the election name will appear on notification page and under

MyElections bar. Max can keep track the election state on this page.

**Scenario 4 Extensions:**

3. Max wants to create another election for a different position of power.

5. Max enters a wrong candidate ID. He is notified to recheck the ID and try again.

**Scenario 5:**

**Scenario name: Edit Election**

**Participant actor instances: Max: Admin of the site**

Flow of events:

1. Max selects an election to edit.

2. The System verifies that the election is not in progress.

3. The System verifies that the election has not passed.

4. Max modifies election information

i. Max modifies election date, name, or other election information.

ii. Max adds / modifies / removes candidate-voter (see associated use cases).

5. Max submits the changes.

6. The System records the changes.

**Scenario 5 Extensions:**

1a. If the election is in progress Max is notified and is exited from Edit Election.

1b. If the election has passed Max is notified and is exited from Edit Election.

5. If Max exits from Create Election. No election information is saved.

6a. If an election already exists on the new date, the election is not saved and Max is notified of

the conflict. They are given the option to modify the election they were trying to edit.

6b. Max has requested that the election be deleted. The System removes all items associated

with the election and then deletes the election.

**Scenario 6:**

**Scenario name: View Election Results**

**Participant actor instances: Berkay: Member of the site**

Flow of events:

1. Berkay logs into the system using his information. He is a voter who wants to see the

finals standings of the election.

2. Berkay selects the election he participated in.

3. The details/content of the election are shown. The leading candidate (i.e. the candidate

who earned the highest number of votes) is shown at the top of the list with the total

number of votes accumulated beside the candidate.

**Scenario 7:**

**Scenario name: View Voting Progress**

**Participant actor instances: Salim: Candidate**

Flow of events:

1. Salim is contesting in an election for the president of the student council in his university. He

logs into the system using his credentials (email and password). He wants to view where his

current standing is in the ongoing elections.

2. He selects the election he is participating in.

3. Along with the other participants he can see his standing and monitor the proceedings of the

elections.

**Scenario 8:**

**Scenario name: Vote**

**Participant actor instances: Can: Voter**

Flow of events:

1. Can wants to take participate in an ongoing election and vote for his choice of candidate. He

follows the steps in Scenarios 1 and 2 to sign-up and log-in respectively.

2. He checks the list of elections he can participate in. He chooses the one of his preferred

choice.

3. The page reveals the list of candidates that are contesting for that election and he found the

one of his choice. Under the photo of the candidate he clicks on the vote button.

4. The system reveals a pop-up asking for Can’s password (for confirmation). He inputs the

password and clicks on confirm.

5. System returns him to the menu with the list of ongoing elections.

6. If he wants to vote again in a different election he follows steps 2 to 4 again.

**Scenario 8 Extensions:**

3. The candidate selected has been either disqualified or candidate withdrew from elections.

Can is told to choose a different election or candidate.

4. Can inputs the wrong password. He is asked to re-enter the password.

6. He decides to vote again. While in step 2, he chooses the same election as previously

selected. System notifies him he is not eligible for that particular election again as he has

already voted previously.

**Scenario 9:**

**Scenario name: Edit Profile**

**Participant actor instances: Gamze: Member of the site**

Flow of events:

1. Gamze is an existing user and she wants to make some modifications to her profile.

2. She logs into the system.

3. She can choose to modify:

- profile picture

- name

- information

4. She makes the change she wants to make and clicks save changes.

**Scenario 10:**

**Scenario name: Log out**

**Participant actor instances: Khaleel: Member of the site**

Flow of events:

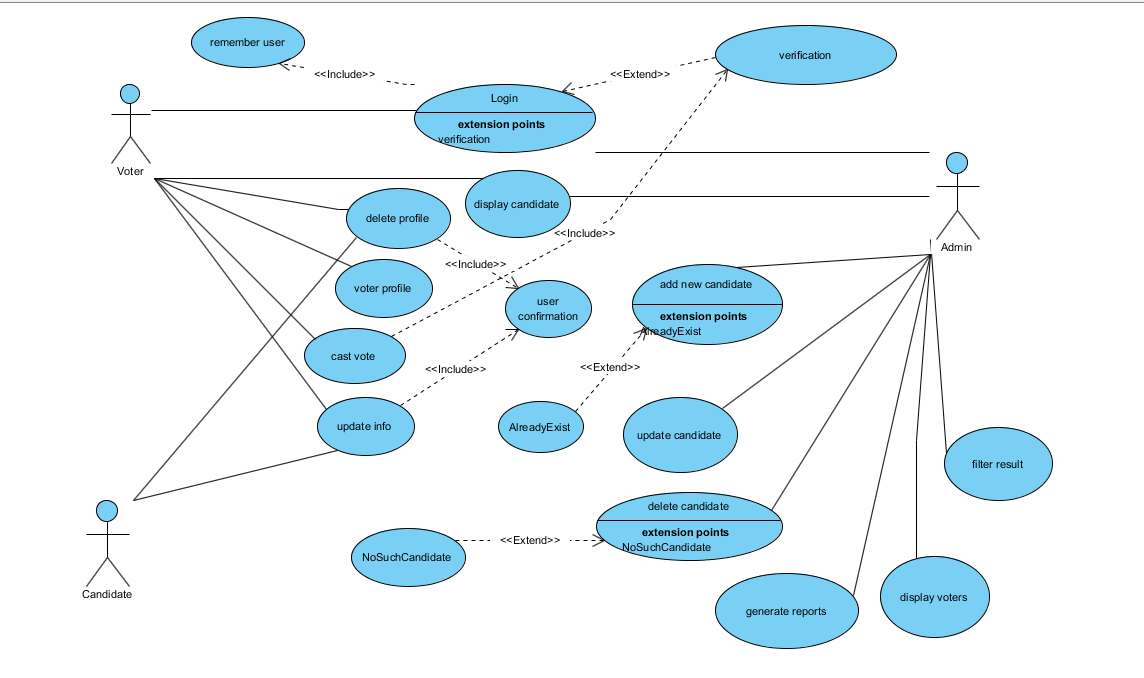
1. Khaleel has just completed voting. He browses through his profile and makes a few changes.

2. He has finished doing all he wants to do and decides to log off.

3. He looks for the log out button at the top right next to his profile and clicks it.

4. System immediately logs him out.

**Use-Case Model**

****

**Use Case 1:**

|  |
| --- |
| *Use case name:*Login  (High Priority) |
| *Participant actors:* Initiated by Voter OR Admin |
| *Flow of events:*  1. The user first enters to Online Election system.  2. The system presents the “Login Form” to the user.  3. The user enters his/her e-mail address into e-mail address text field on the screen, also enters his/her password into password text field on the screen. Lastly, the user sends a request to the ssystem by using login button on the screen to be logged in.  4. The system checks the e-mail addres and password from the “User” table so that the system allows the user to login. Then it determines role of the user by retrieving the role data from the “User” and redirects to proper screen.(Voter or Admin) |
| *Entry Condition:*      The user enters to login screen. |
| *Exit Condition:*         The user is logged in, OR,                                    The user has received an explanation indicating why he/she could not login. |

**Use Case 2:**

|  |
| --- |
| *Use case name:*UpdateInfo  (Low Priority) |
| *Participant actors:* Initiated by Voter, OR Candidate |
| *Flow of events:*  1. The user sends a request to the system to update his/her information which are his/her Identification, photo, Education and Job, current state.  2. Online Election system prepares the “Information Form” by using the “Voter” table or the “Candidate” table. Then, displays it on the screen so that the user can update fields he/she wishes to.  3. The user edits the fields he/she wants, then he/she submits the form by using the update button to be updated.  4. The system updates the information into the “Voter” or the “Candidate” table. Finally, it displays the “Acknowledgement Notice” to the user. |
| *Entry Condition:*     The user is logged into system.                                      The user chooses Update Info option on the screen. |
| *Exit Condition:*        Information of the user has been updated and the user has received an acknowledgement message. OR,                                  The user has received an explanation indicating why he/she could not login. |

**Use Case 3:**

|  |
| --- |
| *Use case name:*Voter Profile (Low Priority) |
| *Participant actors:* Initiated by Voter |
| *Flow of events:*  1. The voter wants to see his profile.  2. The system gets the information about related user by using “Voter” table. |
| *Entry Condition:*      The Voter is logged into online election system.                                   The user chooses My Profile option on the screen. |
| *Exit Condition:*        The Voter has viewed his/her profile content. OR,                                  The Voter has received an explanation indicating why the transaction could not be processed. |

**Use Case 4:**

|  |
| --- |
| *Use case name:*Cast vote  (High Priority) |
| *Participant actors:* Initiated by Voter |
| *Flow of events:*  1. When voter selects cast vote, the system gets the candidates for the related election by using “Candidates” table.  2. The voter selets his/her candidate and clicks Vote button.  3.In order to finish the voting proccess,a verification data needs by the system.The voter fills password text field. |
| *Entry Condition:*      The Voter is logged into the system.                                   The user chooses Cast Vote option on the screen. |
| *Exit Condition:*        The Voter has voted for . OR,                                  The Voter has received an explanation indicating why the transaction could not be processed. |

**Use Case 5:**

|  |
| --- |
| *Use case name:*Delete Profile  (High Priority) |
| *Participant actors:* Initiated by Voter |
| *Flow of events:*1. The Voter sends a request to the system to delete his/her profile on the screen.  2.The system warns the voter and needs user confirmation.  3.The Voter aggrees to delete his profile. |
| *Entry Condition:*      The Voter is logged into the system.                                   The Voter chooses Delete Profile option on the screen. |
| *Exit Condition:*        The Voter has delete his/her profile and log out automaticly. OR,                                  The Voter has received an explanation indicating why the transaction could not be processed. |

**Use Case 6:**

|  |
| --- |
| *Use case name:*Display Candidate  (Low Priority) |
| *Participant actors:* Initiated by Voter OR Admin |
| *Flow of events:*  1. The user sends a request to the system to see candidates on the screen.  2. The system first finds the user whether he/she is an admin or voter.Then lists the candidates for each election for admins OR list the election candidates that the voter related to. |
| *Entry Condition:*      The User is logged into the system.                                  The user chooses Display Candidate option on the screen. |
| *Exit Condition:*        The user has viewed candidates. OR,                                  The user has received an explanation indicating why the transaction could not be processed. |

**Use Case 7:**

|  |
| --- |
| *Use case name:*Display Voters(Low Priority) |
| *Participant actors:* Initiated by Admin |
| *Flow of events:*  1. Admin wants to display all the voters.The system confirms the verification data and pushes voters table.  2. The system lists all voters to the admin. |
| *Entry Condition:*      Admin is logged into online election system.                                   Admin chooses display voters option on the screen. |
| *Exit Condition:*        Admin has displays candidates content. OR,                                  Admin has received an explanation indicating why the transaction could not be processed. |

**Use Case 8:**

|  |
| --- |
| *Use case name:*Generate Reports(High Priority) |
| *Participant actors:* Initiated by Admin |
| *Flow of events:*  1.Admin clicks generate reports and wants to see the results of election assosiated with.  2. The system lists results of the election using election result table. |
| *Entry Condition:*      Admin is logged into online election system.                                   Admin chooses generate reports option on the screen. |
| *Exit Condition:*        Admin has displays result content. OR,                                  Admin has received an explanation indicating why the transaction could not be processed. |

**Use Case 9:**

|  |
| --- |
| *Use case name:*Delete Candidate(High Priority) |
| *Participant actors:* Initiated by Admin |
| *Flow of events:*  1. The system checks the candidate given by using candidate table.  2. Admin clicks delete option on listed candidate. |
| *Entry Condition:*      Admin is logged into online election system.                                   Admin chooses delete candidate option on the screen. |
| *Exit Condition:*        Admin has delete the assosiated candidate and its content. OR,                                   Admin has received an explanation indicating why the transaction could not be processed. |

**Use Case 10:**

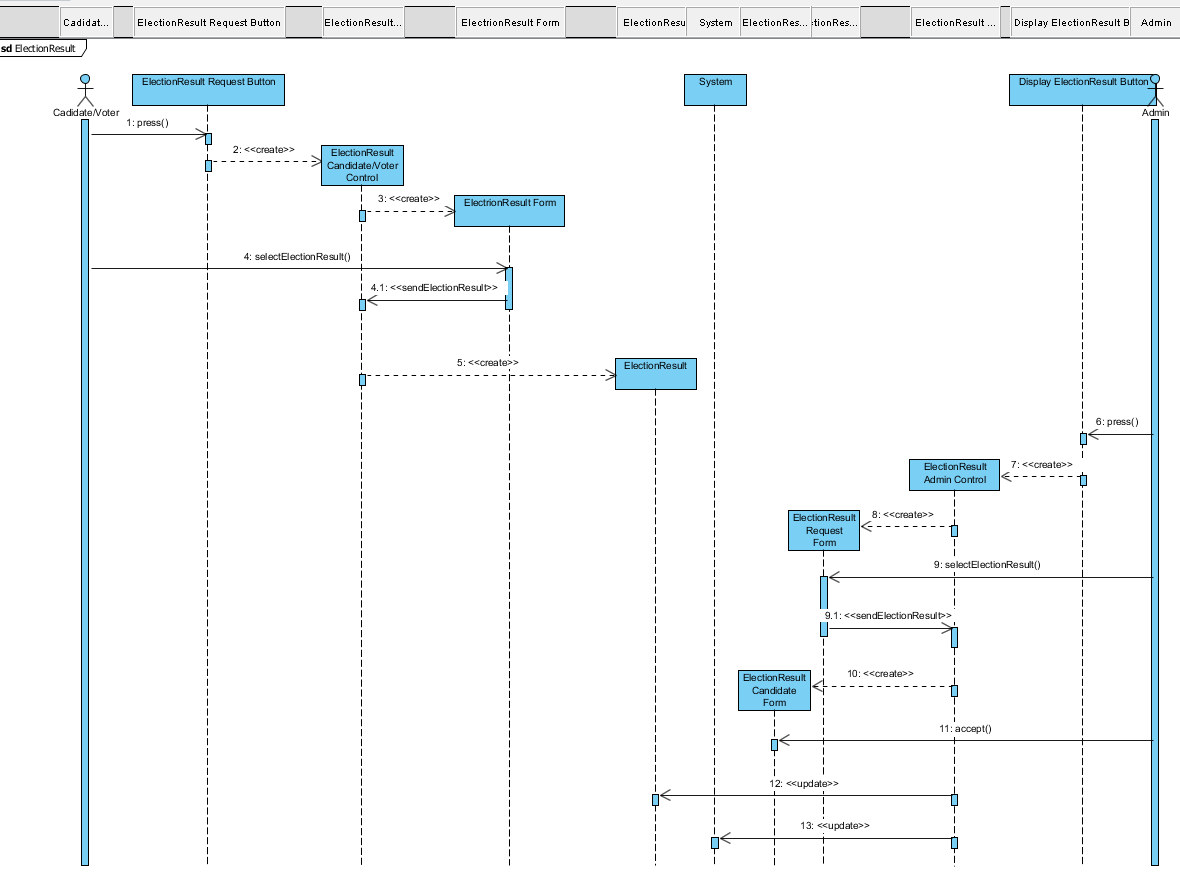
|  |
| --- |
| *Use case name:*Filter Result(High Priority) |
| *Participant actors:* Initiated by Admin |
| *Flow of events:*  1.The system checks genereted reports and lists winning candidates using candidate table.  2. The system lists the candidates that have higher vote. |
| *Entry Condition:*      Admin is logged into online election system.                                   Admin chooses filter result option on the screen.  Reports must be genereted before filtering so the system checks if it is genereted. |
| *Exit Condition:*        Admin has displays filtered results content. OR,                                  Admin has received an explanation indicating why the transaction could not be processed. |

**Use Case 11:**

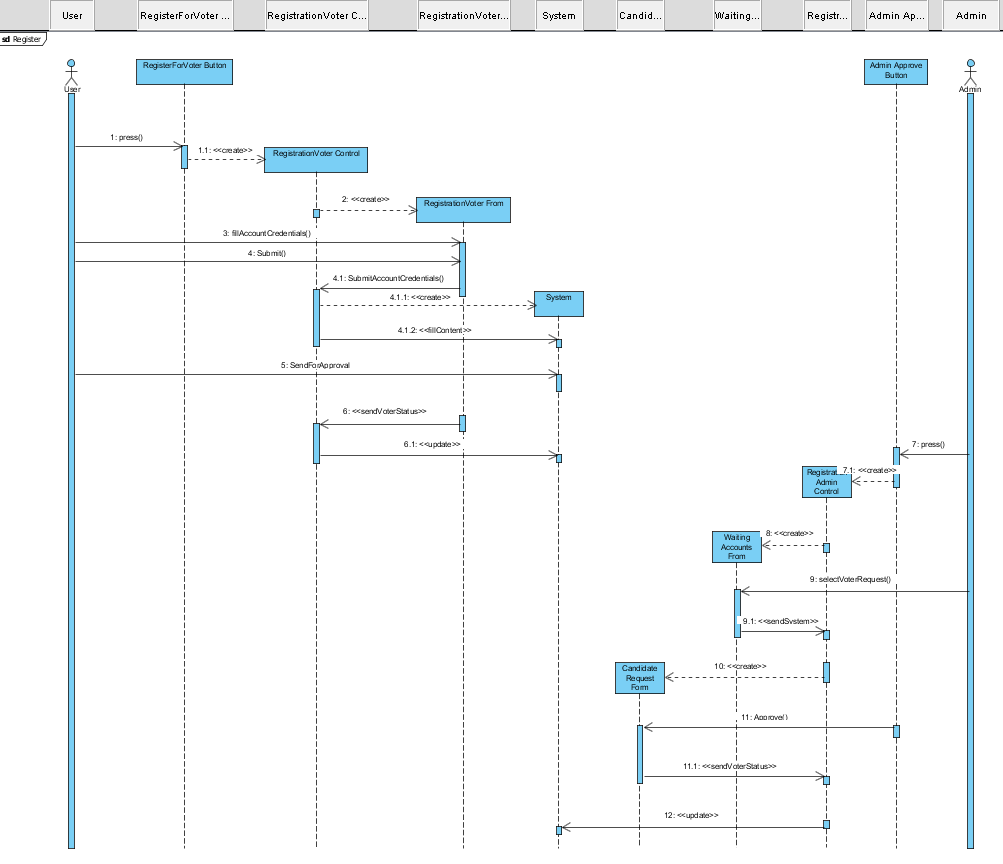
|  |
| --- |
| *Use case name:*Update Candidate(Medium Priority) |
| *Participant actors:* Initiated by Admin |
| *Flow of events:*  1.The system gets the assosiated candidate by using candidate table.  2.The system list the current information form of the candidate.  3.Admin updates needed informations of candidates. |
| *Entry Condition:*      Admin is logged into online election system.                                   Admin chooses update candidate option on the screen. |
| *Exit Condition:*        Admin has update candidate. OR,                                  Admin has received an explanation indicating why the transaction could not be processed. |

**3.4.1 Dynamic Models**

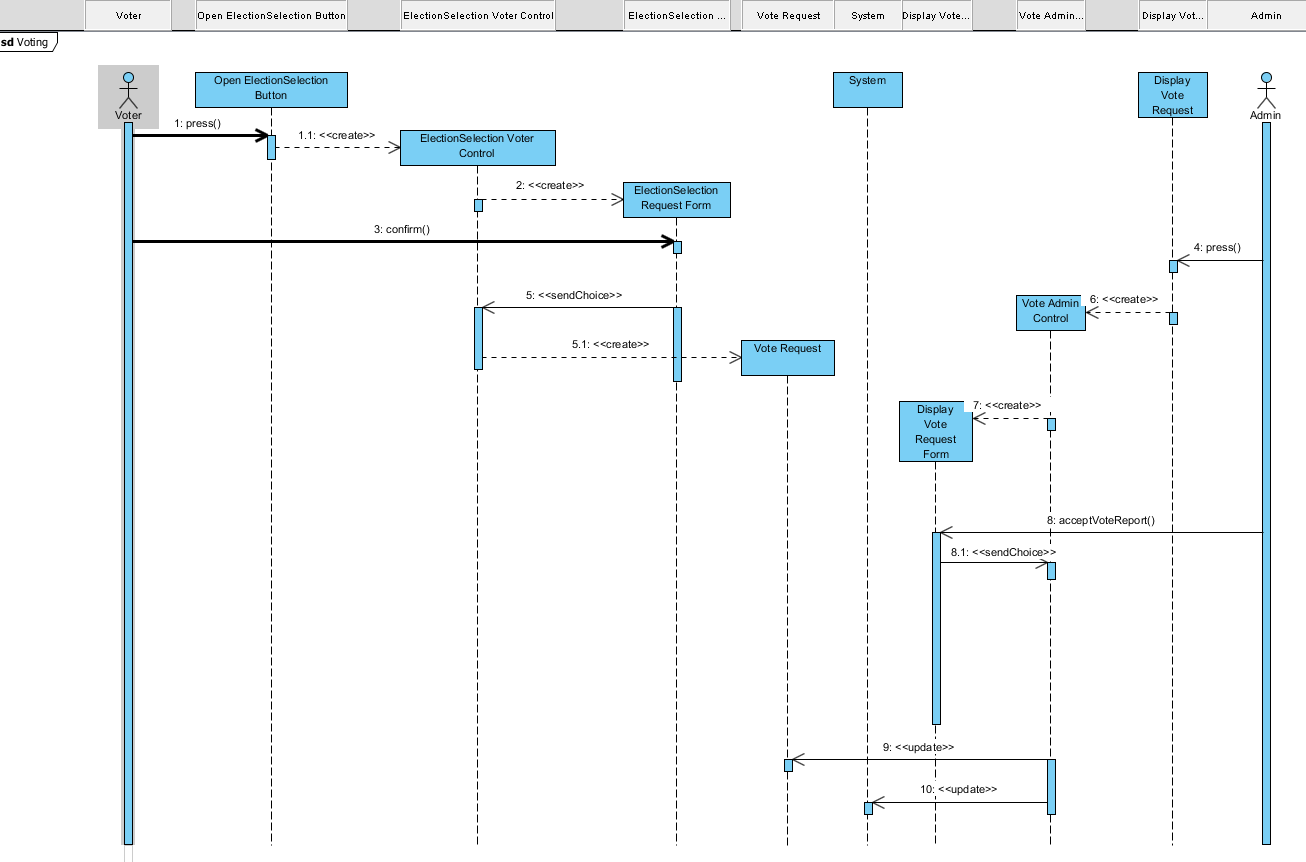
**Election Result- Sequence Diagram**

****

**Register – Sequence Diagram**

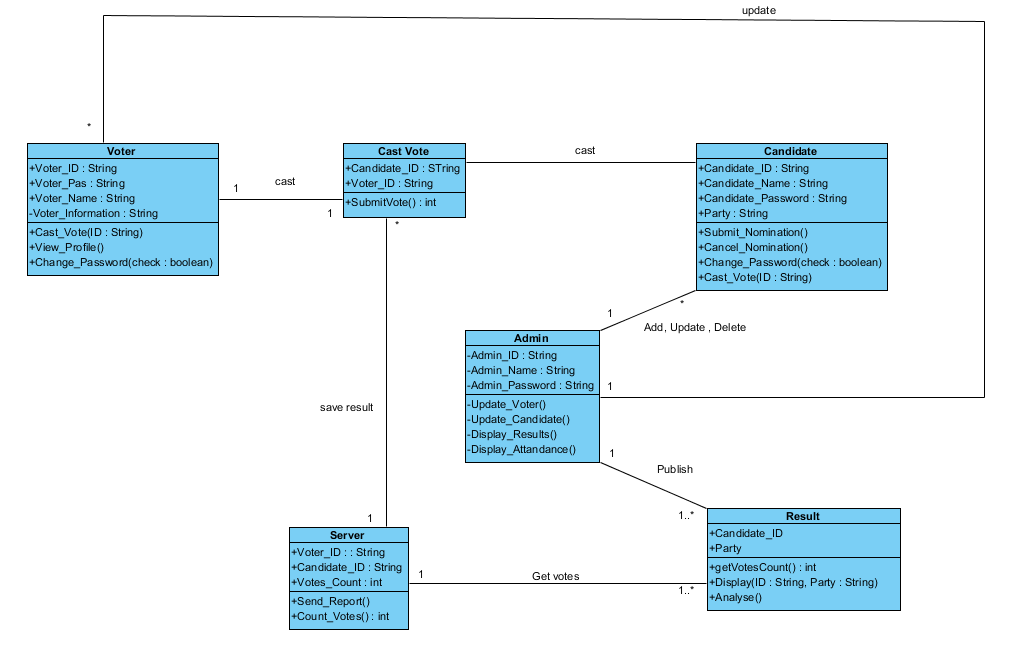
****

**Voting – Sequence Diagram**

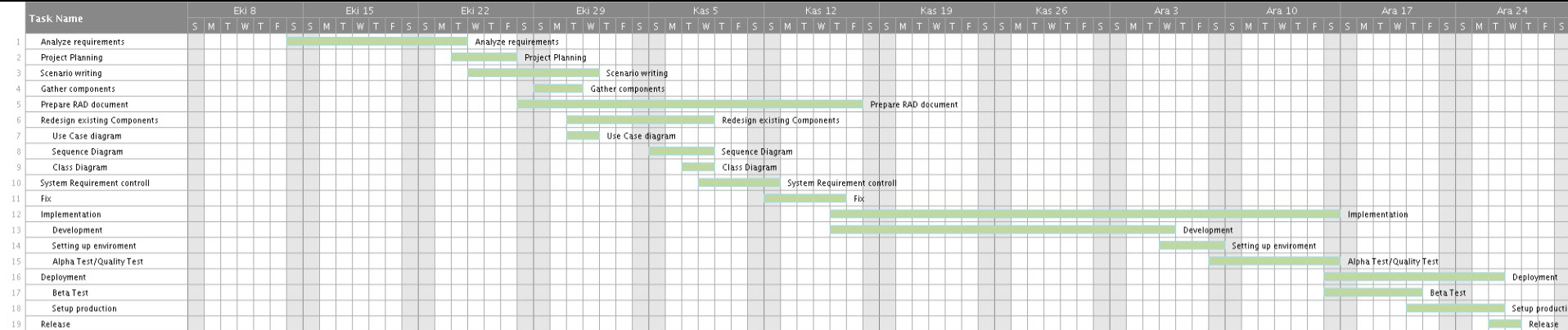
****

**3.4.2 Object Models**

**Class Diagram**

****

**3.5.Project Schedule**



1. **Glossary**

*Voter Table:* Contains voter id , password, name and information.

*Admin Table:* Contains admin id , password and name.

*Candidate Table:* Contains candidate id , name , party and password.

*Cast Vote Table:* Contains candidates Id the votes go for and Voters id that vote for the candidate.

*Result Table:* Contains candidate id and party to show the results*.*

*Server Table:* Contains Voter id , candidate id and count of votes.

*Election Result Form:* Contains the name and party of the election winner candidate and candidates propoganda.

*Election Result Request Form:* Contains the passage to get the spesific elections result,report.

*Election Result Candidate Form:* Contains the passage to get the spesific elections result,report as detailed as admins.

*RegisterVoter Form:* Contains blank places like name , password and information for user to fill in to become a voter.

*Waiting Accounts Form:* Contains non-approved accounts list.

*Candidate Request Form:* Contains name , information and one last blank place for candidates to explain themselves.

*Election Selection Request Form:* Contains the list of avaliable elections.

*Type Error Form:* Contains some information about the possible error types and their explanations which may happen during registration, and their explanation.

*Candidates Form :* Contains the list of candidates

*Display Vote Request Form:* Contains the voter and the vote that gaved to a candidate, contains the validaty of the vote.

*Login Form:* Contains two text fields, one for username and one for password. The form also contains a login button to provide, the functionality of submitting the form, to the user.

*Acknowledgement Notice:* Contains a message which indicates the process has been completed.

*Password Change Form:* Contains three text fields named “password”, “new password”, “confirm new password” and a button named “send” so that the user can submit the form.

*Final Report Form:* Contains the detailed analysis of an election and its winner’s information.

1. **References**
2. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML,* *Patterns, and Java*, Prentice Hall, 3rded.
3. Lecture presentations of the course (the presentations were provided by the Instructor who is Emine Ekin).